

# A Tool for Information Retrieval Research in Peer-to-Peer File Sharing Systems

Linh Thai Nguyen, Wai Gen Yee, Dongmei Jia, Ophir Frieder  
Illinois Institute of Technology  
Chicago, IL 60616, USA  
{linhnt, yee, jia, ophir}@ir.iit.edu

## Abstract

We introduce *IR-Wire*, a tool for information retrieval research and education in peer-to-peer file-sharing systems. Built on top of LimeWire's implementation of the popular Gnutella standard, it includes functionality to collect data on queries and shared files and stores them in a way to make analyses simple. *IR-Wire* is designed modularly to facilitate its customization for other uses.

## 1. Introduction

Peer-to-peer (P2P) file sharing is a popular Internet application with millions of users sharing millions of files daily. Given the scale of the application, it is important that file-sharing systems be efficient. To bolster a better understanding of how such systems operate thus supporting research and development in this area, we introduce *IR-Wire*, a tool for the collection and analyses of data in P2P file-sharing systems. To our knowledge, *IR-Wire* is the only publicly available system that collects data for the file-sharing environment.

*IR-Wire* specifically implements the following features on top of the LimeWire's popular Gnutella client [2]:

1. Information retrieval (IR) style ranking techniques [4].
2. A query logger.
3. A shared data crawler.
4. Tools for data analysis.

Our motivation for building *IR-Wire* is to collect enough statistics on real networks to be able to create realistic models on data distribution and user behavior. Such information has direct bearing on networking, information retrieval, and information security research. For example, it could be applied to the need of creating a standard dataset for P2P IR research [3].

## 2. System description

*IR-Wire* is built on top of LimeWire's open source Gnutella system [2] and written in Java. The architecture of the system is depicted in Figure 1. Embedded in the

basic LimeWire client, *IR+* is a module that implements custom IR functions, such as alternative result ranking techniques. The Data Crawler module collects data that are loaded into a MySQL database that, in turn, is accessed via a Data Analyzer. The Data Analyzer is a separate component to preserve system modularity.

Our modifications made to the core LimeWire system are minimal and are kept in separate, well-defined classes. This reduces the likelihood that independent modifications to LimeWire will affect our system.

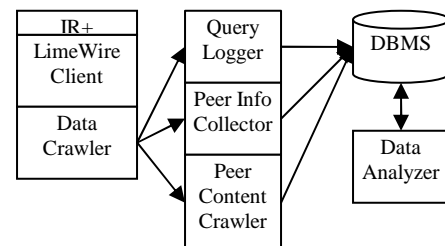


Figure 1. Architectural design of *IR-Wire*

### 2.1. *IR+*

*IR+* contains modifications to LimeWire's query processor and is used for information retrieval research. It implements result-ranking functions and maintains statistics that may improve ranking accuracy. Important statistics, for example, include a count of the number of filenames in which a term occurs and the overall frequency of a term over all filenames.

Two example ranking-functions are *group size* and *cosine similarity* (more ranking functions are described in [4]). *Group size* ranks each result by the number of replicas it has in the result set. This is the standard ranking function in P2P file-sharing systems. *Cosine similarity* maps the query and the filename to a vector space, and ranks each result by the "closeness" of the vectors. *Cosine similarity* is often enhanced with the statistics mentioned above to assign weights to each term in the query.

In Figure 2, we show a screenshot of *IR-Wire*'s search for "Mozart Clarinet." By LimeWire default, *group size* (denoted by "#") is the primary ranking function. Other ranking scores (e.g., *cosine similarity*) are also shown.

#	Name	Size	Bitrate	Term Fr...	Precision	Cosine Si...
37	Mozart Clarinet Quin...	8,808 KB		84	0.289	0.54
35	1-Mozart Clarinet Qu...	6,567 KB		78	0.31	0.576
32	Mozart Clarinet Quin...	8,977 KB		72	0.3	0.56
32	1-Mozart Clarinet Qu...	6,983 KB		82	0.312	0.58
25	1-Mozart Clarinet Co...	8,259 KB		68	0.305	0.569
2	Mozart Symphony C...	6,300 KB	128	2	0.118	0.343
204	Out Of Africa - Cl...	7,252 KB	128	6	0.162	0.477

Figure 2. Screen shot of IR-Wire search results.

## 2.2. Query logger

The query logger records all queries received by a peer in a MySQL table. Each query message contains information including the query terms and type of desired file as described in the Gnutella specification [1].

As per the Gnutella specification [1], the only queries a peer receives are those that it might be able to answer, based on a “content summary” that its neighbors maintain. To retrieve a complete set of queries, we modified the IR-Wire’s content summary to suggest that it shares every possible file.

## 2.3. Peer information collector

We use two techniques to take a census of the peers in the network. The first technique utilizes Gnutella’s network-maintaining heartbeat protocol, called “ping/pong.” Pings are transmitted to peers, which return pongs that contain their IP addresses, port numbers, and network IDs. However, not all peers respond to pings as per the Gnutella specification [1].

Another way we find peers is by issuing “sampling queries,” which are queries that are general enough so that many peers should respond. Query responses also contain information on peers. General queries can be deduced from the query logs.

## 2.4. Peer content crawler

For each discovered peer, our client starts a content-browsing session to collect information about all files shared by that peer. The content-browsing functionality is implemented in LimeWire via the HTTP GET protocol. The most recently discovered peers are crawled first to reduce the failure rate caused by peers leaving the network. For each file, information such as file name, file type, file size, and file’s metadata are put into the database. To expedite the crawling process, we crawl many peers concurrently. Note that peers have the option of disallowing the crawling of their directories.

## 2.5. Data analyzer

The Data Analyzer is a MySQL application and allows the user to perform various analyses through a simple user interface. First, the user is able to write ad-hoc SQL queries to determine information such as the number of peers, the average number of shared files, the identities of these files and the average query length. Secondly, we have implemented a number of more complicated data analyses, which help the user get a better sense of the data. Some analyses are:

- Correlation analysis for the queries: which pairs of keywords frequently co-occur, and which do not.
- Distribution analyses: file popularity distribution, query length distribution, data size distribution, query popularity distribution, and so on.
- Clustering analyses: classification of files according to file type, and the classification of queries according to the query constraints (e.g., type).

We currently have several gigabytes of Gnutella data from the second half of 2006. Sample results obtained include that the average query has 2.94 terms, more than 75% of the queries are for audio files, more than 70% of shared files are audio files, and two highly correlated terms are “hip” and “hop.”

In addition, temporal analyses for the results of all the aforementioned analyses are possible, e.g., how queries or shared data change over time.

## 3. Conclusions

IR-Wire addresses a need in P2P research – a freely available tool that helps collect and analyze data from P2P systems. As a working system, it can also function as a testbed for research on information retrieval, for which we have been using it. To get a copy of IR-Wire and/or collected data, please contact us or visit our Web site, [www.ir.iit.edu/~waigen/proj/pirs](http://www.ir.iit.edu/~waigen/proj/pirs).

## 4. References

- [1] Gnutella Protocol v.0.6. [rfc-gnutella.sourceforge.net/src/rfc-0\\_6-draft.html](http://rfc-gnutella.sourceforge.net/src/rfc-0_6-draft.html), 2002.
- [2] LimeWire home page. <http://www.limewire.com>.
- [3] H. Nottelmann, K. Aberer, J. Callan, and W. Nejdl, The CIKM 2005 Workshop on Information Retrieval in Peer-to-Peer Networks, In *SIGIR Forum*, 40(1), June, 2006.
- [4] W.G. Yee, O. Frieder. On Search in Peer to Peer File Sharing Systems, In *Proc. ACM SAC*, 2005.